**Государственное бюджетное общеобразовательное учреждение "Президентский физико-математический лицей № 239"**

**«Нахождение максимальной площади между параболами»**

Годовой проект по информатике

Автор: Шумилов Алексей, 10-7 класс

Санкт-Петербург

2021

**Постановка задачи**

На плоскости задано множество парабол. Найти такие две параболы, что фигура, находящаяся внутри этих двух парабол замкнута и имеет наибольшую площадь.

В качестве ответа: выделить найденные две параболы, выделить контур фигуры, которая ограничивает точки внутри обеих парабол, желательно выделить внутреннее пространство фигуры («залить цветом»).

Если параболы пересекаются и ограничивают некоторую фигуру, то нужно найти площади фигур между каждой парой парабол, сравнить их и выбрать наибольшую.

**Уточнение исходных и выходных данных**

**Входные данные**

В качестве входных данных выступает список из заданных парабол. Парабола задается координатами своего фокуса и вершины.

Фокус и вершина задаются координатами типа double. Из-за того, что в OpenGL видимая область ограничена, то и координаты ограничены отрезком [-1;1].

**Выходные данные**

В качестве выходных данных выступает пара парабол, площадь между которыми максимальна, и фигура, ограниченная этими параболами. Их нужно выделить.

**Математическая модель**

1. **Коэффициенты параболы через координаты фокуса и вершины**

Парабола задается уравнением: y = a\*x^2 + b\*x + c

Координаты фокуса F.x и F.y, координаты вершины v0.x и v0.y

Из свойств параболы получаем: a = F.y - v0.y;

b = -2\*F.x\*(F.y - v0.y)

c = v0.y + (b^2)/(4\*a)

**2. Координаты точек пересечения парабол**

Одна парабола y1 = a1\*x^2 + b1\*x + c1

Вторая парабола y2 = a2\*x^2 + b2\*x + c2

Приравняв y1 и y2, получим координаты точек пересечения парабол по оси X. Не забываем также поставить условие того, что дискриминант получившегося квадратного уравнения больше нуля либо равен ему, иначе площади между параболами вообще не будет.

Затем подставив эти координаты в уравнение одной из парабол, получим координаты точек пересечения по оси Y.

**3. Площадь между параболами**

Пусть параболы пересекаются в точках x1 и x2, а их коэффициенты равны (a, b, c) с соответствующими индексами. Тогда площадь между ними найдем с помощью определенного интеграла.

Площадь S = |(x2^3 - x1^3)\*(a2-a1)/3 + (x2^2 - x1^2)\*(b2-b1)/2 + (x2 - x1)\*(c2 - c1)|

Модуль стоит потому, что подмодульное выражение может получиться отрицательным, а мы сравниваем площади, как положительные величины.

**Анализ используемой структуры данных**

Нужно хранить список точек–координат фокуса и вершины (входные данные), а также список случайных парабол для того, чтобы их нарисовать. Еще нужно сохранить ссылки на те параболы, которые будут выделены. В процессе я храню значения текущей максимальной площади между параболами, чтобы сравнивать его с вновь полученным. Значение текущей площади не хранится, а значение максимальной–хранится.

**Выбор метода решения**

Для решения задачи я перебираю все пары парабол. Нахожу площадь между каждыми двумя параболами, если найденная площадь между некоторыми параболами больше всех найденных ранее, то запоминаю эту площадь и эти две параболы. После завершения цикла у меня остаются две параболы с наибольшей площадью между ними, и я закрашиваю эти параболы и фигуру между ними.

**Комментированный листинг**

private ArrayList<Point> points;

//список, хранящий координаты фокуса и вершины

private ArrayList<Parabola> parabolas;

//список, хранящий случайные параболы

private ArrayList<Parabola> parabolascolour;

//список, хранящий закрашенные параболы

ParabolaResult parabolaResult;

//результат (две найденные парабол)

public void solve() {

ArrayList<Vector2> vectors;

// перебираем пары точек

double s0 = 0;

//минимально возможное начальное значение площади

Parabola P1 = null;

Parabola P2 = null;

//параболы, которые потом будут найдены и закрашены

for (Parabola p1 : parabolas) {

for (Parabola p2 : parabolas) {

//перебираем все пары парабол

if ((Math.pow((p2.b - p1.b), 2) - 4 \* (p1.a - p2.a) \* (p1.c - p2.c)) > 0) {

//условие того, что дискриминант больше нуля и количество корней равно двум

double x11 = (p2.b - p1.b + Math.sqrt(Math.abs(Math.pow((p2.b - p1.b), 2)) - 4 \* (p1.a - p2.a) \* (p1.c - p2.c))) / (2 \* (p1.a - p2.a));

double x22 = (p2.b - p1.b - Math.sqrt(Math.abs(Math.pow((p2.b - p1.b), 2)) - 4 \* (p1.a - p2.a) \* (p1.c - p2.c))) / (2 \* (p1.a - p2.a));

double y11 = p1.a \* x11 \* x11 + p1.b \* x11 + p1.c;

double y22 = p2.a \* x22 \* x22 + p2.b \* x22 + p2.c;

//нахождение координат точек пересечения парабол

if (Math.abs(x11) < 1 && Math.abs(x22) < 1 && Math.abs(y11) < 1 && Math.abs(y22) < 1) {

//условие того, что точки пересечения лежат в видимой области

double s = Math.abs((Math.pow(x22, 3) - Math.pow(x11, 3)) \* (p2.a - p1.a) / 3 + (Math.pow(x22, 2) - Math.pow(x11, 2)) \* (p2.b - p1.b) / 2 + (x22 - x11) \* (p2.c - p1.c));

//нахождение модуля площади между параболами

if (s > s0) {

//если нынешняя площадь больше предыдущей, то продолжаем

s0 = s;

//максимальной площади присваиваем значения нынешней площади

P1 = p1;

P2 = p2;

//за параболы, площадь между которыми максимальна, принимаем нынешние параболы

parabolascolour.clear();

//удаляем из списка с закрашенными параболами параболы, которые были в нем раньше parabolascolour.add(P1);

parabolascolour.add(P2);

//добавляем в список с заерашенными параболами нынешние параболы

parabolaResult = new ParabolaResult(P1, P2);

//в качестве результата берем вновь найденные параболы

}

}

}

}

}

**Пример работы программы**

Входные данные Выходные данные

(0,80; 0,24)(0,80; 0,76)

(0,52; 0,40)(0,52; 0,14)

(-0,92; 0,11)(-0,92; -0,26)

(0,48; 0,06)(0,48; -0,63)

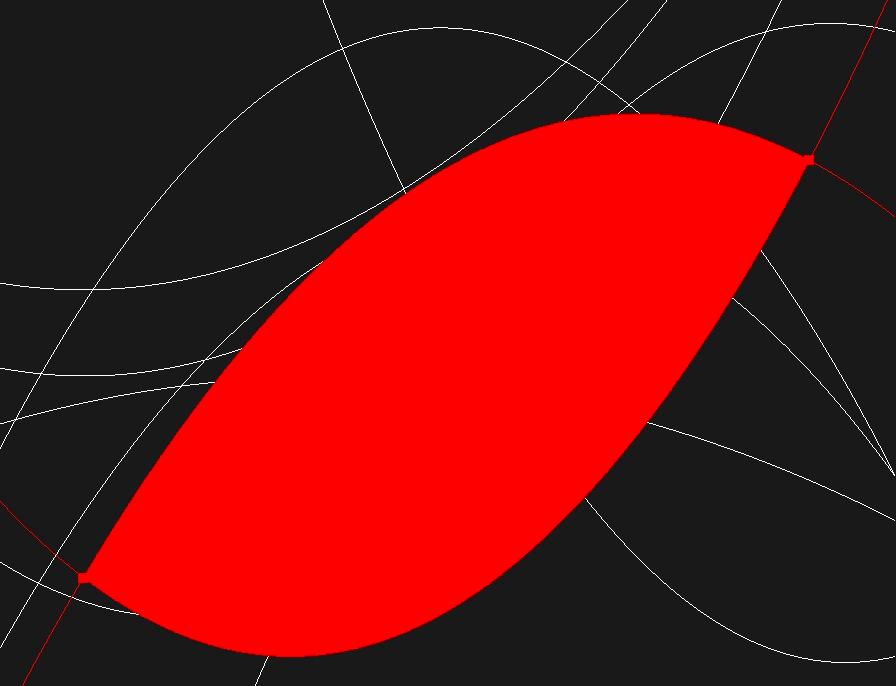
(-0,94; -0,05)(-0,94; 0,35)

(0,70; -0,33) (0,70; -0,54)

(-0,94; -0,05)(-0,94; -0,13)

(0,03; -0,86) (0,03; 0,73)

(-0,17; -0,09) (-0,17; 0,33)

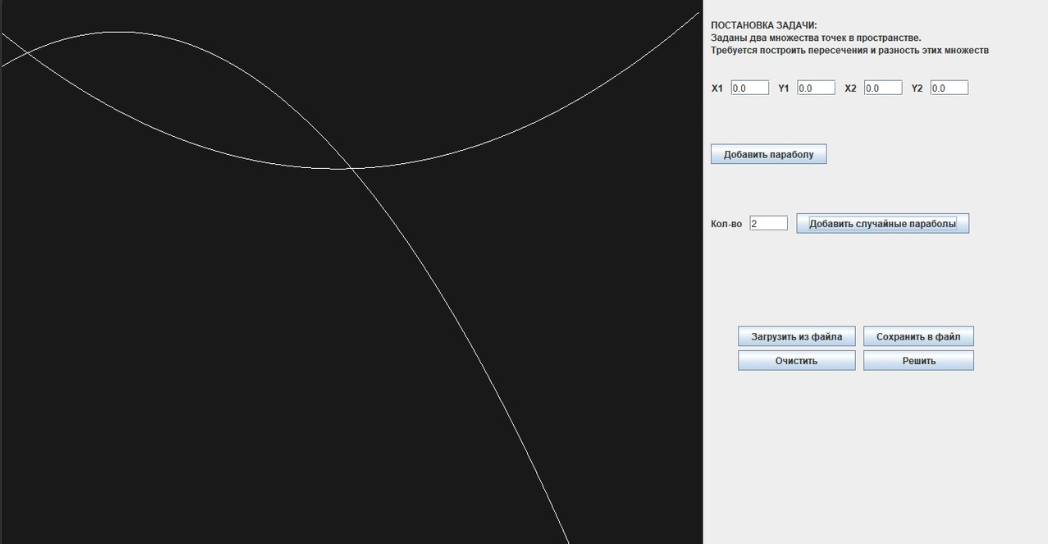
(0,81; 0,56)(0,81; -0,42)

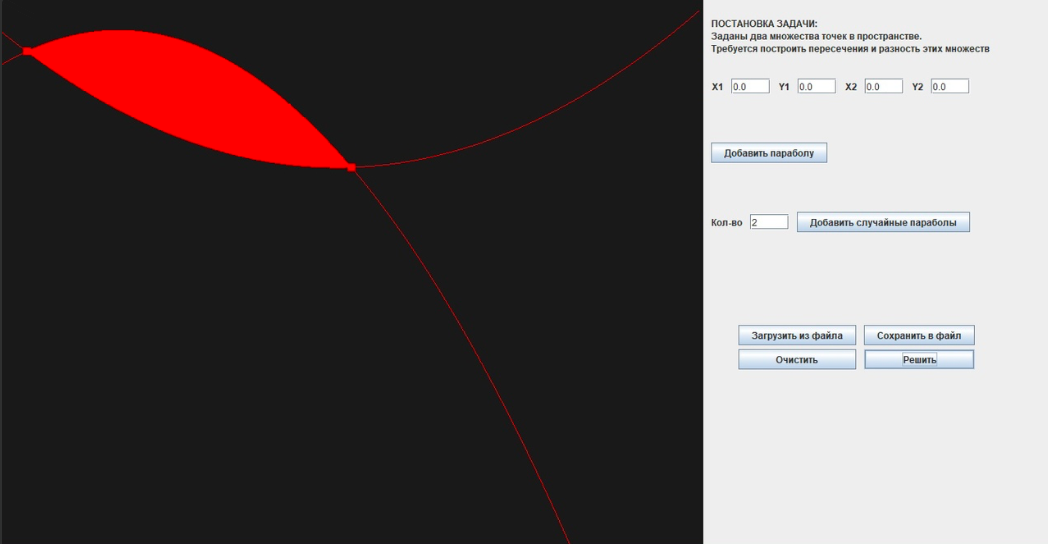
**Анализ правильности решения**

Программа работает правильно, что проверено мною с помощью многочисленных её запусков. Контур фигуры, который выделяется, всегда ограничен именно двумя параболами, а не границами видимой области или как-то ещё.

Также для лишней проверки я выводил максимальное значение площади на экран, и оно всегда оказывалось меньше четырех, даже при очень большом (порядка нескольких тысяч) количестве парабол. Так и должно быть, ведь размеры видимой области в Open GL равняются 2 на 2, а значит площадь этого окошка равна четырем, и было бы по меньшей мере странно, если бы площадь между параболами была больше четырех.

Приведу пример задачи с маленьким количеством парабол, решение которой очевидно, и которое выдает программа.





А в примере ниже параболы не пересекаются, и поэтому программа ничего не выделяет